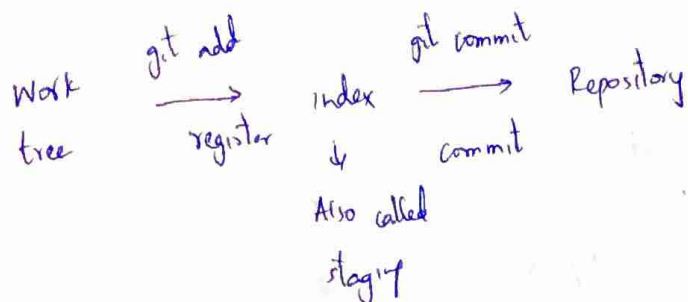


GIT:

May 25, 2017:

Starting off with GIT. Possible to have both local & remote repositories.

Also, possible to create a repository from scratch or copy any existing repository.
(clone)



From the PDF presentation:

- * Commit → 3 parts →
 - a) information about the commit
 - b) Reference to commit that came before (parent commit)
 - c) hash code to identify commit.

- * Branches → Main branch is called the master branch.

→ The head node is the latest commit on the current branch.

→ Branching off an existing branch & merging on to the branch
(~~when~~ procedure used when working on something)

From the web tutorial:

git init to initialize a repository. (delete .git folder to undo)

git status to get the status.

git add <filename> or git add * to move to the staging area (or index).

git commit -m "message" to commit the files into the repository.

⑤

To set the email and name use

```
git config --global user.email "you@example.com"
```

```
git config --global user.name "Your Name"
```

* Create a new repository on github

Choose new repository on ~~github~~ github.com → Add a name

```
git remote add origin <url given by github>
```

(connect local repo to remote repo)

```
git push -u origin master
```

(push changes to remote repo)

} steps are suggested
by github.

So we created a local repository that contained some git documentation & then pushed that onto Github.

* Creating a new branch locally

```
git checkout -b <branch-name> (transfers to a new branch)
```

```
git branch (to check status of branch. Asterisk indicates current branch)
```

```
git checkout master (to switch back into the master branch)
```

So we created a branch called gitgui. (we'll come back to it). We also added a file to our master branch. This file doesn't exist on the gitgui branch (verified using GIT GUI)

So we added a new file to our branch gitgui (not to the master branch)

To push a branch to github :

```
git push origin master.
```

So, now, we have 2 branches → the master & gitgui.

They both have 3 files : 2 of them are common to each other, one is separate.

So, to merge the gitgui branch with the master branch, switch to the master branch
(using `git checkout master`) and then

`git merge gitgui.`

Now the master branch has everything that the gitgui branch has.

We need to delete the gitgui branch locally & remotely.

To locally delete a branch : `git branch -d <branchname>`

To remotely delete a branch : `git push origin --delete <branchname>`

(`git push origin -d <branchname>` also seems to work)